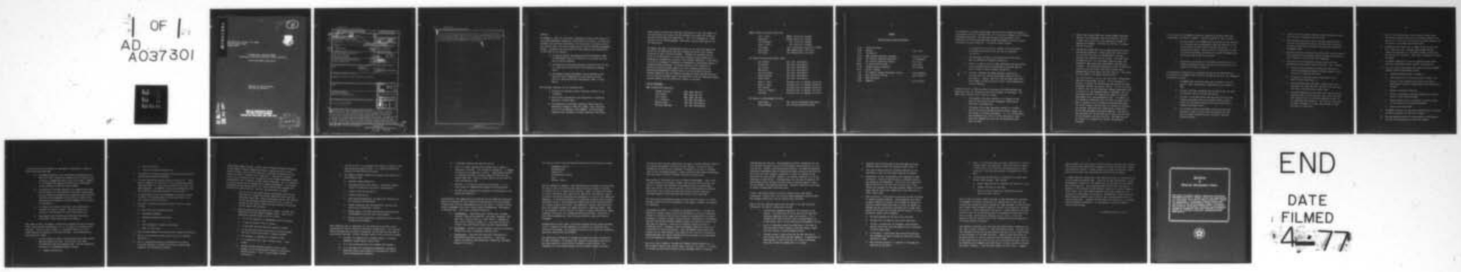
AD-A037 301 SYSTEM DEVELOPMENT CORP SANTA MONICA CALIF F/G 9/2
SOFTWARE DATA COLLECTION STUDY. PROCEEDINGS OF THE DATA COLLECT--ETC(U)
DEC 76 N E WILLMORTH F30602-75-C-0248
UNCLASSIFIED SDC-TM-5542/006/01 RADC-TR-76-329-VOL-6 NL

1 OF 1
AD
A037301



END
DATE
FILMED
4-77

11

RADC-TR-76-329, Volume VI (of eight)
Final Technical Report
December 1976

SOFTWARE DATA COLLECTION STUDY
Proceedings of the Data Collection Problem Conference

System Development Corporation

ROME AIR DEVELOPMENT CENTER
AIR FORCE SYSTEMS COMMAND
GRIFFISS AIR FORCE BASE, NEW YORK 13441

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

19 **REPORT DOCUMENTATION PAGE**

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER 18 RADC-TR-76-329, Vol-[redacted] (of eight) / 2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER 9
4. TITLE (and Subtitle) 6 SOFTWARE DATA COLLECTION STUDY, Proceedings of the Data Collection Problem Conference.	5. TYPE OF REPORT & PERIOD COVERED Final Technical Report, Jun[redacted]75 - Jun[redacted]76
	6. PERFORMING ORG. REPORT NUMBER 14 SDC-TM-5542/006/01
7. AUTHOR(s) 10 N. E. Willmorth	8. CONTRACT OR GRANT NUMBER(s) 15 F30602-75-C-0248
9. PERFORMING ORGANIZATION NAME AND ADDRESS System Development Corporation 2500 Colorado Avenue Santa Monica CA 90406	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 63728F 16 55500810 17 08
11. CONTROLLING OFFICE NAME AND ADDRESS Rome Air Development Center (ISIS) Griffiss AFB NY 13441	12. REPORT DATE 11 Dec[redacted]76
	13. NUMBER OF PAGES 21
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Same 12 24p.	15. SECURITY CLASS. (of this report) UNCLASSIFIED
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE N/A

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

Same

18. SUPPLEMENTARY NOTES

RADC Project Engineer:
Richard T. Slavinski (ISIS)

064
065
5 A036 066

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Data Collection Problems

7- A036 247
8- A036 265

ACCESSION for		
NTIS	White Section	☑
DDC	Buff Section	☐
UNANNOUNCED		☐
JUSTIFICATION........		
BY........		
DISTRIBUTION/AVAILABILITY CODES		
Dist.	AVAIL. and/or SPECIAL	
A		

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A conference was held in December 1975 to discuss the problems of software data collection. Participants were SDC contract personnel associated with one of three software data repositories and the military, civil service, and contract personnel owning or managing these repositories. The repositories included the proposed RADC Software Data Repository, the Space and Missiles Systems Organization's Satellite Control Facility Computer Program Development Library, and the Army Ballistic Missiles Division's Advanced Technology Center Quantitative Data Base. Each repository presented its objectives and described

over

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

339 900

the data collection problems they have encountered. A general discussion of data collection problems ensued, centering around standardization, the reluctance to release sensitive data, and the bias and subjectivity of project reports. No real solutions to the problems were forthcoming but participants left the conference with a better appreciation of the problem.

PREFACE

On December 9, 1975, an invitational conference was held on the premises of the System Development Corporation, Santa Monica, California, to discuss the problems that have been encountered in the collection of accurate, precise and reliable data to be used to manage and to study the software development process. Attendees were associated with three software data repositories in which SDC has an interest:

- A proposed Software Data Repository being developed by Rome Air Development Center for which SDC is on contract to study data collection problems.
- The Quantitative Data Base operated by SDC Huntsville for the U.S. Army Ballistic Missiles Division Advanced Technology Center.
- The Computer Program Development Library operated by SDC Satellite Control Department for the Satellite Control Facility, Space and Missiles Organization, USAF, in Santa Monica.

The principle impetuses for the conference were:

- A paucity of hard data on data collection problems in the literature.
- Difficulties experienced by the repositories in obtaining objective, reliable data.
- The opportunity to exchange information among those just beginning to develop a repository (RADC), whose repository is still in an embryonic stage (BMDATC), and those whose repository has the benefit of mature experience (SCF CPDL).

Those invited to attend the conference consisted of, first, the "owners" of the repositories and, second, the SDC employees associated with operating, planning for and using the repositories. The list of attendees as shown on pages 3 and 4 include managers, technicians, advisers and suppliers for the repositories.

The agenda (see page 5) included descriptions of the intent and operations of each of the repositories and their problems, and closed with a general discussion of data collection problems. Although the conference did not result in the derivation of any definite solutions to the problems, nor in an exhaustive consideration in depth of the problems themselves, it did result in an active exchange of information and engendered a considerable amount of thought provoking discussion. Focusing attention on the difficulty of acquiring valid and reliable data to be used in managing projects and performing methodological research serves to bring the data collection problems themselves into our research programs, and raises the hope that ways will be found to eliminate much of the subjectivity and bias that have plagued software productivity and reliability research in the past.

LIST OF PARTIPANTS

RADC Software Data Repository

Richard Slavinski	RADC, Rome, New York
John Palaimo	RADC, Rome, New York
Rocco Iuorno	RADC, Rome, New York
N. E. Willmorth	SDC, R&D, Santa Monica
Marcia Finfer	SDC, R&D, Santa Monica
Marjorie Templeton	SDC, R&D, Santa Monica

BMDATC Software Technology Repository

Carl C. Davis	BMDATC, Huntsville, Alabama
Buddy Dace	BMDATC, Huntsville, Alabama
Iver Bakkegard	SDC, Huntsville, Alabama
Robert Corelli	SDC, Huntsville, Alabama
John Lawson	Texas Instruments, Huntsville, Alabama
Barry Boehm	TRW, Redondo Beach, California
Tom Thayer	TRW, Redondo Beach, California

SCF Computer Program Development Library

Jerry Hansen	SDC, CPIC, Santa Monica
Milt Winsor	SDC, CPIC, Santa Monica
Lee Tillman	SDC, CPIC, Santa Monica
Bob Shapiro	SDC, CPIC, Santa Monica
John R. Orlando	SDC, CPIC, Santa Monica
Peter Armerding	SDC, CPDL, Santa Monica
Margo Dragoo	SDC, CPDL, Santa Monica
Robert E. Berri	Aerospace Corp, El Segundo, California
Bruce L. Adams	Aerospace Corp, El Segundo, California
Earl Ragland	Aerospace Corp, El Segundo, California
Francis J. Zampino	Aerospace Corp, El Segundo, California
Alan Gott	Aerospace Corp, El Segundo, California

SDC Research and Development Division

Terry Court	SDC, Software Development Department
Harvey Bratman	SDC, Sofware Engineering Branch

AGENDA

Software Data Collection Problems

8:30	Coffee and Donuts	
9:00	Welcome	Terry Court
9:10	Introduction	
9:20	RADC Software Repository Concepts	Richard Slavinski
9:45	ATC Software Repository Operations	Iver Bakkegard
10:25	ATC Software Technology Program	Carl Davis
10:50	TI Data Collection Study	John Lawson
11:30	Software System Integration	Jerry Hansen
12:15	Buffet Luncheon	
1:15	SCF Computer Program Development Library	Peter Armerding
2:00	Data Collection Problems	Gus Willmorth
2:30	Open Discussion	
4:30	Summation	Gus Willmorth

The Conference on Software Development Data Collection Problems was opened by G. Willmorth of SDC on 9 December 1975. Mr. Terry Court, Manager of the Software Development Department, SDC, welcomed the guests and presented a brief summation of problems associated with software development directly relating to productivity and reliability. The points he made include:

1. The recognition by the entire industry of the seriousness of software unreliability and the need for improving reliability.
2. The techniques and tools in existence do not consistently meet the problems of software reliability.
3. There is a need to know the right combination and applications of the available tools and techniques to software development in order to improve program reliability.
4. In order to perform the proper analyses, data must be collected from the software development process to aid in the analysis. However, there are time and money constraints which hamper the data collection effort; these problems must also be addressed.

Richard Slavinski of RADC presented an overview of the RADC Software Data Repository after the introduction of all participants of conference. The major points covered by Mr. Slavinski include:

1. The purpose of the data repository is a response to the needs of data acquisition and management identified by contractors and symposiums.

2. The concept of the data repository consists of a central facility for data collected during the design, coding, and test phases of software development. It is to include output from automated tools, such as utility and analysis tools. There will probably be a facility for desensitizing the data collected.

3. Some of the problems RADC sees include common terminology necessary for correlation, data classification, security of data, and flexibility of the repository to adapt to changing environments. Of these, flexibility is probably the key to success.

4. Two parallel studies were let to study the data collection and the data repository. These studies will form a base for the pilot facility to be established in the year following the completion of the studies. The study being performed by SDC is to investigate production data, the problems of data acquisition requirements, and data base structure. The study being performed by IITRI is addressing security, data base specs, application program specs, documentation library specs, and pilot facility. The reason for two studies is to obtain optimum results.

5. The pilot facility is to be a test bed, with a limited data base and limited number of users. It will function as a nucleus, or center, for software data and analyses for both management functions and quality control. It will be used as a research tool for new technology; language development tool for analysis of new language features; management tool for developing baselines for comparative studies for software development costs; and documentation tool. In summary, it will become an interactive tool, flexible to support future work in research, cost and schedule estimation techniques. The primary emphasis was previously on reliability analysis, but concepts involving costs, productivity, and maintainability have enlarged the original emphasis. The tools of the repository will be available to all users, but this will not be included in the pilot facility. This facility will be limited in its capabilities and will be constantly evaluated.

Dr. Carl Davis of the BMDATC presented an overview of the data repository established in Huntsville, Alabama. The points made by Dr. Davis include:

1. The major thrust of the BMDATC data collection and data analysis is to evaluate the new programming techniques employed in the research work being performed by the ARC contractors. It is a multi-contract effort. Data collection is being done in order to improve requirements specification, program development and verification and validation techniques.

2. The analysis of the data will ultimately provide data on the software development process, the data collection process, and quality and reliability metrics.

Iver Bakkegard of SDC-Huntsville presented the work he was responsible for associated with the BMDATC Data Collection and Analysis project. Mr. Bakkegard's presentation included:

1. The BMDATC Data Collection effort was initiated in the 3rd quarter, 1974, and represents approximately 15-18 months of data.

2. The data collection procedures were written by SDC, but they incorporated inputs by the ARC contractors.

3. The goals of the Quantitative Data Base were to assess the software development efforts and support tools, provide visability into project productivity, and collect data on software development costs.

4. The different software development projects were briefly reviewed in order to acquaint the participants with the type of complex research projects involved in the data collection effort.

5. Three of the four projects reached a stable maintenance state, after which no further data was collected.

6. Work had been initiated in two of the large programs before the data collection effort began, which may have contributed to some of the problems incurred in obtaining and evaluating development data.

7. One data collection problem was the three month reporting period which perhaps was not frequent enough, although the contractors were internally collecting data more frequently.

8. A discussion of the reporting forms was given, with some observations on the effectivity of the forms. This included:

   a. Difficulty with accurately tracking and reporting of types of programming statements without automatic tools to assist in reporting this data.

   b. Collection of data must have minimum impact on the people who are submitting it.

   c. Every contractor had individual, internal forms and procedures for reporting data; the result was that they then had to translate data to other forms.

   d. A Software Modification Data Acquisition form was to be submitted when an error occurred. Only errors that took more than one day to correct were reported in order to not overburden the contractor. This form was first used to report "breakage." (It was later decided to report all errors on this form.)

9. Some of the estimated data, reflecting work accomplished before data collection project was initiated, covered a two-year period. Since this is the bulk of the data obtained, few conclusions can be drawn from the data base.

10. Rescheduling and redirection is common in the research work area. Because of this, previous code becomes obsolete. Productivity measures (if calculated as number instructions per man day) are "alarming". Much code is produced, but later discarded.

11. The BMDATC software has a level of complexity much higher than usual compounding normal problems with complexity.

12. A list of some of the problems that have surfaced with BMDATC research software development include:

    a. Language differences (POL's and MOL's.)

    b. Cost accounting for multi-activity/multi-project runs

    c. Volumes of code scrapped due to poorly anticipated requirements (coding to "pseudo-specs") and experimental failures.

    d. Changes in technical direction.

    e. Failure to develop satisfactory productivity measures (which is still being examined).

    f. Unsatisfactory results in attempts to minimize impact of data collection on program development.

    g. Costs of data collection.

13. The BMDATC personnel are considering modifications to the data collection procedure for their next attempt.

14. The data gathered to date is being examined in the hopes of proving a more optimum way of producing software.

In the discussion that followed, Mr. Bakkegard's presentation, a number of interesting points were made.

- Tom Thayer pointed out that the problems encountered are not peculiar to R&D software but commonly occur on projects such as avionics, command and control and civil systems. However, it is conceded that changes of direction (and requirements) is more prevalent in R&D than other systems.
- The incentive to the ARC contractors for submitting the data was the same as for any contractor; the means of enforcement of data collection procedures was through the contract monitor. There must be encouragement to contractors to submit valid, unbiased data, followed by a positive feedback of the results of the analysis.
- A customer can dictate all one wants, but a contractor is reluctant to report on himself. One must convince the contractor that the data will not be used against him.
- Requirements specifications may be the most significant contributor to the final productivity measure.

John Lawson of Texas Instruments in Huntsville presented the data collection effort instituted semi-automatically at T.I. in support of the Quantitative Data Base. They also want to obtain: 1) estimation factors, 2) error significance by the analysis of the data collected. The following points were made by Mr. Lawson:

1. The data needed to support the objectives were product quality data and product-cost data. The principle data items collected were labor and computer usage. The data supporting the analysis of management problems include:

    a. Budgeting/forecasting

b. Status reporting

c. Resource consumption/productivity

2. The design of the control documents for obtaining data should be done before program development.

3. All work reported was against a WBS on a daily basis. There were 10 categories of activities; in the top-down development approach used by T.I., work was concurrently being done in all work categories. (The BMDATC Process Design Methodology does not follow the traditional software life-cycle model.)

4. It was possible to track the development of a module through 17 steps. The largest number of errors were found in unit test, usually using test drivers.

5. The factors impacting the quantification of work to be done include:

   a. Quality of specs/interface control

   b. Programming language

   c. Storage utilization construction

   d. I/O

   e. Size of program (in object instructions)

   f. "New" vs "old" code

6. Labor estimate depends on productivity plus degree of difficulty.

7. Type of instructions per module may be one way to quantify difficulty.

8. The T.I. productivity measure as determined by the data collected indicated in extremely high-rate, exceeding SAGE, Safeguard, and Site Defense Software.

Jerry Hansen, Deputy Director of SDC's Satellite Control Program and manager of the Computer Program Integration Contract, reported on the Air Force Satellite Control Facility (SCF) data base, collected and maintained by SCF. The SCF has been in existence for fifteen years. They have a voluminous library, but the amount of real information on the software development cannot be determined since no analysis of that type has been done. SDC is responsible for collecting data on the second half of software development - the software integration and maintenance effort. Mr. Hansen familiarized attendees with the Satellite Control Facility. Briefly, the presentation included:

1. The facility is the ground support environment system for DoD R&D Satellite Systems and has been supporting multi-satellite operations since 1962. Several contractors provide the technical arm for developing and maintaining the facility in order to provide mission control staffs with data required for satellite vehicle control and evaluation.

2. Data are processed by two operational systems: the Real Time System and the Flight Support Computer System. Some of the operational problems faced by the SCF include:

   a. The facility supports 17 independent satellite programs.

   b. It uses 50 computers from six vendors.

   c. It uses both real-time and batch computing systems.

   d. It has over two million instructions in the ops program and four million instructions in support programs.

   e. The software is produced by 16 contractor teams.

   f. It is operational full time - 24 hours a day - 7 days a week.

   g. Perhaps the most important problem is that, on the average, 1/3 of the operational software is modified or replaced per year. (This is approximately 700,000 instructions.)

3. The CPIC mission is to guarantee the integrity of AFSCF in the data processing support system. It is used for operational control of AF R&D satellite missions.

4. The computer program integration performed by SDC consists of the following tasks:

   a. Detailed system engineering.

   b. Interface definition and control - (including responsibility for contractors having well defined specs.)

   c. Product review and evaluation.

   d. Production monitoring.

   e. Data system documentation, including the integration of all contractors' documentation.

   f. System integration test and evaluation, including the monitoring of contractors' testing.

   g. System support, including the providing and maintaining of development facilities and liaison support.

   h. Control of system evaluation, including the formal control of modifications/changes.

Peter Armerding, who is responsible for the operation of the CPDL, gave a presentation on the SCF repository following a luncheon break. This talk was mainly concerned with the flow of information into the CPDL and the kinds of data therein contained. Briefly, the major points of the discussion included:

1. The CPDL is a repository for software products, including documents, program masters, data blocks.

2. It is the distribution point for documents and programs.

3. It is the center for configuration management, designed to ensure the quality and integrity of the product by configuration accounting and recording.

4. It provides technical and clerical service.

5. There is a formal change/control process when a product is delivered to the CPDL. This consists of rulings on: a) changes to design; b) discrepancies via ECP's. Approximately 40 ECP's are received per month; the total number of forms received numbers about 400. There are status reports on activities for every configuration item.

6. The CPDL is a library for tracking of satellites. It is a focal center for communications between interested parties.

7. The CPDL could provide error discovery or tracking data if one wanted that capability.

Gus Willmorth of SDC summarized the information that the RADC Data Collection Study has extracted from the literature concerning data collection problems. Software data collection requirements exist on two levels - a process control (project management) level and a quality control (methodology improvement) or research level. The classes of data collected are:

- Environmental: Application area, contract type, customer relations, resource availability - personnel, equipment, software tools, physical facility, stress factors - adequacy of time, skills, manning, storage, etc., and stability factors - turnover rate, modification rates, and other uncertainties.

- Performance: Planned vs actual schedules, resource utilization, productivity rates, and product characteristics.

- Configuration: Functional and structural characteristics, modification statistics, error statistics, -- abilities figures (reliability, maintainability, operability, and other measures of quality.)

The classes of data collection problems derived from the literature included:

Management Conflict
Standardization
Subjectivity
Instrumentation Effects
Costs
Systemic Effects

Not only workmen but management and corporations are reluctant to provide data. One doesn't want to hurt the project monitor by being a bearer of bad news nor offend him into a "head rolling" reaction. There is a natural reluctance to release information that causes one to look bad or lose face, and a desire to protect proprietary methodology and techniques so as not to lose competitive advantage. If the monitor employs or is perceived to employ coercive and/or threatening tactics, counter-aggressive behavior may result and be justified as a proper response to a threat. Non-compliance, evasion, falsification of data, and sabotage attempts are common reactions to perceived threat. Some of the management conflict reflects resistance to change, even when the change procedures are easier, less threatening and more efficient. Resistance is also often accompanied by claims of excessive effort, both justified and unfounded.

Standardization of data items, collection procedures and of project characteristics is needed to provide comparability of measures in evaluting tools, techniques and methods. We are gradually moving in this direction; repository operations should hasten the movement.

The subjectivity of measures is perhaps the largest source of unreliability for data collection. In the past, most research cross-projects have had to rely upon subjective, after-the-fact estimates of what occurred. The intangibility of the software process and the software product are often

cited as the basis of much subjectivity, but ways to achieve greater visibility for software development are now fairly well defined. An overall lack of information and the failure to generate it whether due to costs or laziness leads to many uninformed estimates. Worse, estimates often remain uncorrected even after better information becomes available to adjust them.

Many sources of subjective bias exist in software development. Some of this is sheer optimism, a reluctance to admit that anything could go wrong, and some of it results from resistance to perceived threat as covered above. Other bias results from internal politics, biases and prejudices concerning applications, tools, people and what-have-you, and from the pleasant or unpleasant impacts of past experience and history of the individual.

All these biases and prejudices come into play distorting data as it filters through successive levels of management, being summed, averaged, and selected for reporting.

Instrumentation effects -- also known as Heisenberg effects -- include the behaviorial changes, process delays and interferences, and other distortions created in a process by the very act of observing and measuring the process. Resentment, irritation, greater caution and care, forgetting and interference and sheer time delays providing progress reports and preparing briefings are some negative responses. Some positive responses, generally known as "Hawthorne Effects", frequently result from people knowing that they are in an experiment or are being observed. Such improvements in motivation and productivity are great enough to cast doubts on any claim for a technique or methodology where the results obtained with the technique in an experimental trial are compared to "industry norms."

Any action that is taken to increase the fineness of data granularity -- a more frequent sampling rate, greater depth, precision, or detail -- is likely to increase not only direct dollar cost but secondary losses in time and

interferences with the work. More automated collection techniques will not only increase the objectivity of data but decrease data granularity at a low cost per bit. However, developing project monitors, instrumenting operating systems and programming tools and developing product evaluation and verification tools are costly projects, especially if these are required of and maintained for every project, computer, and supplier regardless of size and complexity of applications. Teleprocessing, too, can ease and speed up the data gathering task, but that too costs money in terminal devices, transmission channels, and computer processing. Hopefully, benefits can be found to offset increased data collection costs.

Finally, the normal problems of control systems such as time delays, asynchronies, instabilities, and failures plague management control systems and introduce distortions into the software data collected.

Some of the more important points that were made in the open discussion following Dr. Willmorth's presentation include:

1. Configuration management depends on the resources of the project. The traditional Air Force position has been that they want something at the lowest cost possible. It is estimated that data collection costs are approximately 3% of the project costs for the configuration management's office.
2. Levels of importance should be established in the kinds of data one should collect because of the huge amount of data one could collect in the development process.
3. The most success in data collection has been realized in those places where there has been feedback. A generalized raw data approach to data collection with analysis performed later may relieve the burden early in a project.

4. Automatic data collection may be the only means to ensure objective data but short term projects cannot afford it.

5. The costs of data collection ultimately come back to the government as they are the largest procurer of software.

6. Researchers, as well as developers, are interested in data collection. A data collection clause in software contracts may eventually be commonplace. The initial cost figures for data collection will be high, but will diminish in time. Perhaps the customer will motivate software vendors to collect data, with the result that competition in the field will in effect reduce costs.

7. Data collection is very possible; analyzing the data collected is another problem altogether. [There appeared to be almost an equal difference in opinion as to whether the analysis or objective should dictate what data to collect or whether to proceed with the collection of available data in hopes that analysis of the data will provide fruitful results.] In this discussion, several points emerged:

   a. One must parameterize the data to be collected.

   b. One must know the specific use of each data point.

   c. The data collected must be thoroughly verfied before analysis (The error rate and degree of bias is believed to be seriously high.)

   d. If the data is insufficient when the desired analyses are performed, one should, expand the data base to meet the specific needs.

   e. More efficient analysis is possible if one knows the objective of the thrust.

8. There is a definitive need to provide a definition of terms to provide a basis for comparison. There must also be a degree in discipline in the collection and catagorization of data collection. Also, one must account for subjectivity in the data.

9. In discussing the reluctance of contractors to submit data, several ideas emerged. They include:

   a. Remove repercussions to contractor for telling the truth.

   b. Remove collection of cost data.

   c. Maintain direct contact with contractor and obtain data first hand from him.

Also, in the discussions, Boehm and Thayer of TRW reaffirmed the reluctance of management to release information, but reported that on more than one instance even though the project manager was ready and eager to provide openly and in detail much information about his project, the customer was reluctant to receive it. Others agreed that this was so. Project monitors often have several projects to oversee and can easily be swamped with data unless they have the proper intrastructure necessary to sort it out. In the SCF community, SDC CPIC performs much of this interpretive function abetted by the System Engineer, Aerospace.

Earl Ragland of Aerospace said that what the project monitor needed was not more data, but more information -- the distillation of data. Slavinski said that information extraction through modeling and analytic tools would be one of the prime goals of the RADC repository. Shapiro of SDC stated that he believed strongly in "seat-of-the-pants" decision making; that the function of the system should be to deliver the facts to the decision-maker, who with his experience and judgment could often come to a faster and better solution than could a computer. There was some support for this notion -- computer

decision models are usually gross simplifications of the real-life situation which may have many unquantified parameters including the political climate and personality factors. Further, people will accept arbitrariness from a human decision-maker that they would not tolerate from a computer.

It seemed generally agreed that it was impossible to get all subjectivity out of the software development data. There are too many uncertainties in the developmental situation and the complexities are too great for easy comprehension and modeling. Our forecasting models should take into account measures of the uncertainties, and forecasts should be in terms of ranges of values (time, costs, performance characteristics) and probabilities of occurrence or achievement, not absolutes. Nevertheless, we should seek to define objective measures and to look for and create measurable events to improve predictions. Breaking the work down into smaller units - creating micro-milestones and products -- is one approach to achieving greater precision and accuracy.

☆U.S. GOVERNMENT PRINTING OFFICE: 1977-714-025/88

MISSION
of
Rome Air Development Center

RADC plans and conducts research, exploratory and advanced development programs in command, control, and communications (C^3) activities, and in the C^3 areas of information sciences and intelligence. The principal technical mission areas are communications, electromagnetic guidance and control, surveillance of ground and aerospace objects, intelligence data collection and handling, information system technology, ionospheric propagation, solid state sciences, microwave physics and electronic reliability, maintainability and compatibility.

AMERICAN REVOLUTION BICENTENNIAL 1776-1976